



AR 20

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Chandran, et al.

Attorney Docket No.: CISC100/1156

Application No.: 09/276,917

Examiner: NGUYEN, DUSTIN

Filed: March 26, 1999

Group: 2154

Title: NETWORK TRAFFIC SHAPING USING
TIME-BASED QUEUES

Confirmation No.: 2820

01/23/2008 SSITHIB1 00000081 09276917

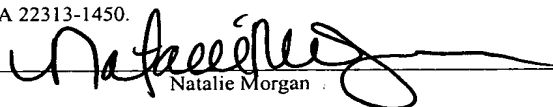
02 FC:1255

2230.00 OP

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the U.S. Postal Service with sufficient postage as first-class mail on January 14, 2008 in an envelope addressed to the Commissioner for Patents, P.O. Box 1450 Alexandria, VA 22313-1450.

Signed:


Natalie Morgan

**APPEAL BRIEF TRANSMITTAL
(37 CFR 192)**

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This brief is in furtherance of the Notice of Appeal filed in this case on June 13, 2007.

This application is on behalf of

☐

Small Entity

☒

Large Entity

Pursuant to 37 CFR 1.17(f), the fee for filing the Appeal Brief is:

☐

\$255.00 (Small Entity)

☒

\$510.00 (Large Entity)

☒

Applicant(s) hereby petition for a Five month extension(s) of time to under 37 CFR 1.136.

If an additional extension of time is required, please consider this a petition therefor.

☐

\$ An extension for _____ months has already been secured and the fee paid therefor of _____ is deducted from the total fee due for the total months of extension now requested.

☒

Applicant(s) believe that no (additional) Extension of Time is required; however, if it is determined that such an extension is required, Applicant(s) hereby petition that such an

extension be granted and authorize the Commissioner to charge the required fees for an Extension of Time under 37 CFR 1.136 to Deposit Account No. 500388.

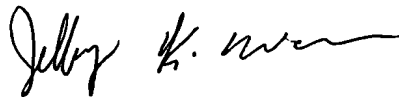
Total Fee Due:

Appeal Brief fee	\$510.00
Extension Fee (if any)	\$2,230.00
Total Fee Due	\$2,740.00

☒ Enclosed is Check No. 14137 in the amount of \$2,740.00

☒ The Commissioner is authorized to charge any additional fees or credit any overpayment to Deposit Account No. 500388, (Order No. CISC100).

Respectfully submitted,
BEYER WEAVER LLP



Jeffrey K. Weaver
Reg. No. 31,314

P.O. Box 70250
Oakland, CA 94612-0250
(510) 663-1100



PATENT

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

EX PARTE CHANDRAN *et al.*

Application for Patent

Filed: March 26, 1999

Serial No. 09/276,917

FOR:

NETWORK TRAFFIC SHAPING USING TIME-BASED QUEUES

APPEAL BRIEF

01/23/2008 SSITHIB1 00000081 09276917

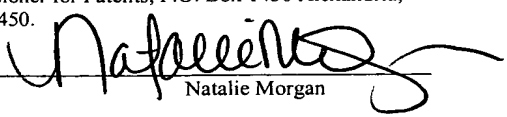
01 FC:1402

510.00 OP

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the U.S. Postal Service with sufficient postage as first-class mail on January 14, 2008 in an envelope addressed to the Commissioner for Patents, P.O. Box 1450 Alexandria, VA 22313-1450.

Signed: _____


Natalie Morgan

Art unit: 2154

Examiner: Nguyen, Dustin

TABLE OF CONTENTS

	<u>Page No.</u>
1. REAL PARTY IN INTEREST	1
2. RELATED APPEALS AND INTERFERENCES	1
3. STATUS OF CLAIMS	1
4. STATUS OF AMENDMENTS	1
5. SUMMARY OF CLAIMED SUBJECT MATTER	2
6. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL	10
7. ARGUMENT	11
8. CLAIMS APPENDIX	19
9. EVIDENCE APPENDIX	24
10. RELATED PROCEEDINGS APPENDIX	25



(1) REAL PARTY IN INTEREST

The real party in interest is Cisco Technology, Inc., the assignee of record.

Address: 170 West Tasman Drive, San Jose, CA 95134-1706

(2) RELATED APPEALS AND INTERFERENCES

N/A

(3) STATUS OF CLAIMS

There are a total of 29 claims pending in this application (claims 1-28, and 30). Claim 29 has been cancelled. Claims 1-28, and 30 were examined and rejected. No claims have been allowed.

Claims 1-28, and 30 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over US Patent No. 6,389,019 issued May 14, 2002 to Fan et al. (hereinafter, "Fan") in view of US Patent No. 6,621,792 issued September 16, 2003 to Petty (hereinafter, "Petty"). There are no other rejections. The rejection of each of claims 1-28, and 30 under § 103(a) is appealed.

(4) STATUS OF AMENDMENTS

No amendment has been filed in response to the final rejection mailed on March 16, 2007.

(5) SUMMARY OF CLAIMED SUBJECT MATTER

The present invention is directed to apparatus and methods for controlling data flow through a network using a plurality of time-based queues. Specifically, it relates to a time-based queuing system including a series of queues, each of which buffers packets for a defined length of time.

The claimed invention focuses on six embodiments of a time-based queuing system presented by independent claims 1, 12, 15, 25, 28, and 30.

Independent claim 1

Claim 1, one of the six independent claims on appeal, recites the following features in addition to a memory and one or more processors:

a plurality of time-based queues logically configured on the memory and together defining a period of time with each time-based queue defining a separate increment of time within the period of time, whereby each time-based queue is set to dequeue all of its contents at a separate time, every time that a specified increment of time elapses.

Claim 1 further recites that

“the processor is configured or designed to direct (i) data or (ii) grants to transmit data to particular time-based queues based upon network traffic shaping delays prescribed for the data or grants to transmit the data,” and

“each time-based queue is configured to dequeue more than one packet.”

Aspects of the present invention provide a time-based buffering system that buffers network data (e.g., packets) based upon how long the data should be held in order to comply with a traffic shaping policy. The data’s source or destination need not be considered – although it can be – in determining how and where to buffer the data.

Typically the time-based buffering system includes a collection of time-based queues (sometimes referred to as “buckets”), each of which is scheduled for a different time to dequeue. An example of such buckets is illustrated as the queuing structure 18 in Figure 1.

The system controlling traffic shaping may determine how long a particular piece of data should be buffered (a “traffic shaping delay”) until it can be put on the

network. Then, based upon that length of time, the system chooses one of the time-based queues in which to buffer the data. That chosen queue has a dequeuing time that matches the traffic shaping delay. Queue identifications may be recycled, so that after a chosen queue dequeues its contents (at the specified time), it is rescheduled for a new dequeuing time. It is then available to buffer new data that must be delayed by a time matching the new dequeuing time.

The plurality of time-based queues of claim 1 is illustrated, for example, as the structure 18 (B_{M-1} through B_5) in Figure 1, and is described in detail at page 7, lines 5-26; and page 13, lines 1-17. As illustrated in Figure 4, the system dequeues, at 407, all those packets in queues that have timed-out since the last dequeuing event. Figure 1 shows more than one packet (actually 2 packets) being dequeued from queue B_4 . See the discussion at page 7, lines 14-17.

The memory and processors of claim 1 are described at page 4, lines 4-6 and Figure 9, block 950 for example.

Dependent claims 2-11

Each of claims 2-11 depends directly from claim 1.

Claim 2 specifies that “the apparatus is a router.” This claim is described, for example, at page 21, lines 1-3. See also Figure 10.

Claim 3 recites that “the apparatus is a cable modem termination system.” This feature is described, for example, at page 14, lines 4-6 and Figure 5A, item 504.

Claim 4 is directed to an embodiment where the separate increments of time defined by the time-based queues are each of the same length. This is described, for example, at page 6, lines 1-2.

Claim 5 is directed to an embodiment where the separate increments of time defined by the time-based queues are configurable. This is described, for example, at page 8, line 22 - page 9, line 17. See also page 16, lines 25-31.

Claim 6 specifies that the period of time defined by the plurality of time-based queues is configurable. This is described, for example, at page 8, line 22 to page 9, line 17.

Claim 7 recites that the one or more processors are further configured or designed to determine network traffic shaping delay. This claim is discussed, for example, at page 11, line 20 - page 12, line 7. See also block 215 of Figure 2.

Claim 8 is directed to an embodiment where the one or more processors are further configured or designed to discard data or a request to grant transmission of data if a network traffic shaping delay is greater than the period of time defined by the plurality of time-based queues. Support for this claim is found, for example, at page 17, lines 11-18. See also steps 217 and 219 of Figure 2 as well as the arrow to the right of the “rate policer” block 14 in Figure 1.

Claim 9 specifies that the one or more processors are further configured or designed to transmit, without buffering in a time-based queue, the data or issue grants to transmit data if there is no network traffic shaping delay. This is described, for example, at page 4, lines 12-19. See also the arrow to the left of “rate policer” block 14 in Figure 1 and operation 213 of Figure 2.

Claim 10 recites that the one or more processors are further configured or designed to direct network packets of varying sizes to the time-based queues. This claim is discussed, for example, at page 2, lines 18-20.

Claim 11 specifies that the apparatus is configured or designed to simultaneously buffer, in a single time-based queue, data or grants to transmit data from a plurality of network nodes. This is described, for example, at page 4, lines 20-25.

Independent claim 12

An independent claim 12 recites an apparatus for controlling data flow through a network. Claim 12 specifically recites the following:

traffic shaping means for determining how long to buffer data or grants to transmit data; and

buffering means for buffering the data or grants to transmit data in a plurality of time-based queues together defining a period of time, with each time-based queue defining a separate increment of time within the period of time, whereby each time-based queue is set to dequeue all of its contents at a separate time, every time that a specified increment of time elapses.

Claim 12 further recites that “each time-based queue is configured to dequeue more than one packet.”

In some ways, this claim corresponds to claim 1, and thus, is supported by at least the description and the figures referred to in connection with claim 1. See specifically block 16 of Figure 1 (and page 8, lines 8-21) as an example of the traffic shaping means. Block 221 and related description also present information about the

traffic shaping means. The buffering means of claim 12 is presented in Figure 1, feature 18, for example, as well as the discussion beginning on page 7, line 5 et seq. See also page 10, lines 10-21.

Dependent claims 13-14

Each of claims 13-14 depends directly from claim 12.

Claim 13 specifies that the traffic shaping means also directs the data or grant to transmit data to particular time-based queues based upon a determined length of time for buffering. This claim is described, for example, at page 11, lines 1-12. see also block 221 of Figure 2 and the discussion at page 8, line 22 to page 9, line 3 and page 12, lines 8-13.

Claim 14 recites that a policing means determines whether to buffer the data or grants to transmit data. This is described, for example, at page 6, lines 24-30. See also block 14 of Figure 1 and blocks 211, 213, and 215 of Figure 2.

Independent claim 15

An independent claim 15 specifies a method of controlling data flow through a network and corresponds in some ways to claim 1. Claim 15 specifically recites the following:

- determining that transmitting additional data to or from a network node will breach a policy for the network node;
- selecting one of a plurality of time-based queues that together define a period of time, with each time-based queue defining a separate increment of time within the time period, whereby each time-based queue is set to dequeue all of its contents at a separate time associated with its increment of time, every time that a specified increment of time elapses; and
- buffering the additional data or a grant to transmit the additional data in the selected one of the plurality of time-based queues.

Further, claim 15 recites that “at least some times when de-queuing of all its contents, a time-based queue dequeues more than one packet.”

The first operation (“determining that transmitting additional data to or from a network node will breach a policy for the network node”) is described at decision block 611 of Figure 2 as well as page 6, line 32 to page 7, line 2 and page 7, line 27 to page 8, line 1.

The second and third operations (“selecting one of a plurality of time-based queues that together define a period of time” and “buffering the additional data or a grant”) are discussed at page 8, lines 8-14 and shown at blocks 221 and 223 of Figure 2 for example. The plurality of time-based queues is shown as item 18 in Figure 1 and the dequeuing and buffering operations are depicted in Figures 3 and 4, for example.

Aspects of claim 15 correspond to claim 1, and thus, are supported in part by at least the description and the figures referred to in connection with claim 1.

Dependent claims 16-24

Each of claims 16-24 depends directly or indirectly from claim 15.

Claim 16 recites “receiving data addressed to the network node prior to determining that transmitting additional data will breach the policy,” and that “the data addressed to the network node is the additional data.” This feature is described, for example, at page 2, line 21- page 3, line 2.

Claim 17 recites “receiving data sent by the network node prior to determining that transmitting the additional data will breach the policy,” and that “the data sent by the network node is the additional data.” This claim is described at, for example, page 2, line 21- page 3, line 2.

Claim 18 recites “calculating a network capacity used by the network node if the additional data was to be transmitted, the calculation being performed prior to determining that transmitting the additional data will breach the policy.” This claim is described at, for example, page 2, line 21- page 3, line 2.

Claim 19 recites “determining a delay until the additional data can be transmitted,” and that “the determined delay is used to select the time-based queue.” This claim is supported, for example, at page 3, line 7 - page 4, line 19. See also Figure 2, blocks 215 and 221.

Claim 20 recites that “the time-based queue is selected by matching its time to dequeue with the delay determined for the additional data.” This claim is supported, for example, at page 3, lines 7 to 15.

Claim 21 recites “dequeuing the additional data; and transmitting the additional data without exceeding the maximum allowed data flow for the network.” This claim is supported, for example, at page 9, lines 18-34.

Claim 22 recites “receiving new data that does not form part of the additional data; determining that transmitting the new data will breach the policy; determining a

delay until the new data can be transmitted without exceeding the maximum allowed data flow for the network node; and determining that the delay is sufficiently long that the new data is discarded without buffering in the time-based queues.” These features are depicted in, for example, the loop and associated operations triggered at blocks 207 and 209 of Figure 2. See also the discussion of Figure 2 at page 11, line 1 to page 12, line 21.

Claim 23 recites that “the separate increments of time defined by the time-based queues are each of the same size.” This is described, for example, at page 6, lines 1-2.

Claim 24 recites that “the increments of time defined by the time-based queues are configurable,” and that “the period of time defined by the plurality of time-based queues is configurable.” This is described, for example, at page 8, line 22 - page 9, line 17. See also page 16, lines 25-31.

Independent claim 25

An independent claim 25 specifies a particular embodiment (i.e., a computer program product) of a method of controlling data flow through a network recited in claim 15. Specifically, claim 25 recites “[a] computer program product comprising a machine-readable medium on which are stored program instructions for controlling data flow through a network.” The claimed program instructions includes the following:

“determining that transmitting additional data to or from a network node will breach a policy for the network node;
selecting one of a plurality of time-based queues that together define a period of time, with each time-based queue defining a separate increment of time within the time period, whereby each time-based queue is set to dequeue all of its contents at a separate time associated with its increment of time, every time that a specified increment of time elapses; and
buffering the additional data or a grant to transmit the additional data in the selected one of the plurality of time-based queues.”

Further, claim 25 recites that “at least some times when de-queuing of all its contents, a time-based queue dequeues more than one packet.”

The instructions of claim 25 correspond to the method operations of claim 15, and thus, claim 25 is supported by at least the description and the figures referred to in connection with claim 15. Further, the machine readable medium is depicted in

elements 957 of Figure 9 and 1061 of Figure 10 and described at page 23, lines 1-20, and page 21, lines 14-19.

Dependent claims 26-27

Each of claims 26-27 depends directly from claim 25.

Claim 26 recites “program instructions for calculating a network capacity used by the network node if the additional data was to be transmitted, the calculation being performed prior to determining that transmitting the additional data will breach the policy.” Claim 26 corresponds to claim 18. This claim is described at, for example, page 2, line 21- page 3, line 2.

Claim 27 recites program instructions for:

- receiving new data that does not form part of the additional data;
- determining that transmitting the new data will breach the policy;
- determining a delay until the new data can be transmitted without exceeding the maximum allowed data flow for the network node; and
- determining that the delay is sufficiently long that the new data is discarded without buffering in the time-based queues.

Claim 27 corresponds to claim 22. These features are depicted in, for example, the loop and associated operations triggered at blocks 207 and 209 of Figure 2. See also the discussion of Figure 2 at page 11, line 1 to page 12, line 21.

Independent claim 28

An independent claim 28 specifies a particular embodiment (i.e., a computer program product) of a method of controlling data flow through a network recited in claim 15. Specifically, claim 25 recites “[a] computer program product comprising a machine-readable medium on which is provided program instructions for controlling data flow through a network.” The claimed program instructions include the following:

- program code for determining that transmitting additional data to or from a network node will breach a policy for the network node;
- program code for selecting one of a plurality of time-based queues that together define a period of time, with each

time-based queue defining a separate increment of time within the time period, whereby each time-based queue is set to dequeue all of its contents at a separate time associated with its increment of time, every time that a specified increment of time elapses; and

program code for buffering the additional data or a grant to transmit the additional data in the selected one of the plurality of time-based queues.”

Further, claim 28 recites that “at least some times when dequeuing of all its contents, a time-based queue dequeues more than one packet.

The program code of claim 28 correspond to the method operations of claim 15, and thus, claim 28 is supported by at least the description and the figures referred to in connection with claims 15 and 25.

Independent claim 30

An independent claim 30 recites an apparatus for controlling data flow through a network. Specifically, claim 30 recites the following:

means for determining that transmitting additional data to or from a network node will breach a policy for the network node;

means for selecting one of a plurality of time-based queues that together define a period of time, with each time-based queue defining a separate increment of time within the time period, whereby each time-based queue is set to dequeue all of its contents at a separate time associated with its increment of time, every time that a specified increment of time elapses; and

means for buffering the additional data or a grant to transmit the additional data in the selected one of the plurality of time-based queues.

Claim 30 further recites that “each time-based queue is configured to dequeue more than one packet.”

The functions of claim 30 correspond to the method operations of claim 15, and thus, are supported by at least the description and the figures referred to in connection with claim 15. Examples of suitable structures for performing these functions are presented at the same locations and in Figures 1 (a logical representation of each means element), Figure 5 (particularly block 504) and Figures 9 and 10 (blocks 950 and 1062) as well as the text descriptions of these features, which were identified above.

(6) GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

There is one ground of rejection to be reviewed on appeal:

Claims 1-28, and 30 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over a combination of the Fan et al. and Petty patents.

The claims do not stand or fall together and some will be argued separately.

Specifically, claims 1, 2, 4-8, 10-13, 15, 19-25, 28, and 30 will be argued as one group. Claim 3 will be argued separately. Claim 9 will be argued separately. Claim 14 will be argued separately. Claims 16 and 17 will be argued as a separate group. Finally, claims 18 and 26 will be argued as a separate group.

(7) ARGUMENT

REJECTIONS OF CLAIMS 1-28, AND 30 UNDER 35 U.S.C. § 103(a)

Introduction

Claims 1-28, and 30 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over a combination of Fan and Petty.

Each appealed claim is separately patentable and the patentability of each will be argued separately as set forth in Section (6). The Appellants' explanation of the differences between the above-cited references and the claimed invention will first be discussed for claim 1, and then for the dependent claims, each of which recites an additional limitation not found or suggested in the context of the cited references. Independent claims 12, 15, 25, 28, and 30 and their dependent claims will be similarly discussed.

Independent claim 1

Claim 1 recites, *inter alia*, the following:

“a plurality of time-based queues logically configured on the memory and together defining a period of time with each time-based queue defining a separate increment of time within the period of time, *whereby each time-based queue is set to dequeue all of its contents at a separate time, every time that a specified increment of time elapses.*” (Emphasis added)

Claim 1 further recites that “the processor is configured or designed to direct (i) data or (ii) grants to transmit data to particular time-based queues based upon network traffic shaping delays prescribed for the data or grants to transmit the data,” and that “each time-based queue is configured to dequeue *more than one packet.*” (Emphasis added) Thus, the queues have the ability to hold and dequeue more than one packet.

As admitted by the Examiner in the Final Office Action dated March 16, 2007, the Fan patent does not disclose at least one of the above-identified claimed features, namely, “each time-based queue is configured to dequeue more than one packet.” In order to cure the deficiency of the Fan patent, the Examiner cited the Petty patent. Appellants respectfully traverse.

Appellants have previously pointed out that in Petty the queues dequeue only a single cell at one time. Petty's system, like Fan's system, is focused on ATM systems where each cell is the same size and cells are constructed at a particular rate. Nothing in Petty suggests dequeuing more than one cell at a time from a queue. By contrast the invention of claim 1 offers considerably more flexibility, which is appropriate for handling cells of various sizes as is the norm in TCP/IP.

Despite the Examiner's position that Petty's ATM queues can and do dequeue multiple ATM cells at any given time, a careful and complete reading of the specification does not support this view. The Petty specification consistently refers to enqueueing and dequeuing only a single cell at a time.

Petty describes an ATM cell constructor 100 having a traffic shaper 121 comprised of an enqueue state machine 140, a dequeue state machine 141, and a plurality of queues 130. See Figure 1. The traffic shaper 121 is presented as a replacement for conventional algorithms for dequeuing, which are described in Petty's background section. As described by Petty, ATM standards provide a Generic Cell Rate Algorithm (GCRA) for use by traffic shapers for meeting out cells. See column 2, lines 36-60. As discussed there, the GCRA requires the output queue of each ATM virtual circuit to be processed in a particular way.

The queues of all virtual circuits are processed during each period. If the queue of a virtual circuit is empty, the traffic shaper does not transmit a cell from that queue. If a cell is available in the queue of a virtual circuit, the traffic shaper transmits it if it meets the GCRA criterion, i.e., if the period between the last transmitted cell from this queue and now is greater than or equal to the minimum administered cell-rate period. *If more than one cell is available in the queue of a virtual circuit, the traffic shaper transmits one cell from the queue if it meets the GCRA criterion, and the next cell must wait at least until the subsequent processing period to be transmitted.* A spacing of at least 125 us between transmitted ATM cells from any virtual circuit is thus ensured, thereby guaranteeing that the virtual circuit does not exceed its negotiated peak cell rate. See column 2, lines 45-60 (emphasis added).

Petty describes a type of traffic shaper in which multiple queues are shared by multiple virtual circuits. At any given instant in time, a given queue is responsible for a particular virtual circuit. If a cell from a particular virtual circuit is received too late to be queued in its primary designated queue, the cell is placed in a different queue

and the overall queue responsibility is shifted. This process is described generally at column 3, lines 7-53 as well as Figure 2 and the associated discussion.

The Examiner's Position

In the Response to Arguments section at page 2 of the Action dated March 16, 2007, the Action cites – in this order – column 9, lines 52-62 (and Figure 10); column 6, lines 2-5; and column 8, lines 23-28 of the Petty patent. These sections are said to support the Examiner's position that Petty discloses "dequeuing more than one packet as claimed." However, these portions fail to teach or suggest that any queue is "set to dequeue all of its contents at a separate time" and that such contents may be "more than one packet."

The cited sections pertain to dequeuing from multiple queues. Of course, more than one packet/cell will be dequeued from multiple queues, but this has nothing to do with dequeuing multiple packets from a single queue as claimed.

Second, the Action cites column 7, lines 40-60; and column 9, lines 52-62 of Petty. Specifically, the Action points to the description of the dequeuing function in these sections. Column 7, lines 40-42; and column 9, lines 52-53 of the Petty patent respectively contain the following nearly identical text:

A dequeing [sic] function of FIG. 6 dequeues cells from one of the shaper queues 131 during each processing interval of 125 μ s.

The function of FIG. 10 dequeues cells from one of the shaper queues 131 during each cell-processing period.

It appears that the drafter of the specification used a cut and paste function (with minor editing) to generate the second of these two sentences. Regardless of how the sentences were drafted, each one is consistent with an interpretation in which a single cell is dequeued per each dequeuing interval. The use of the plural "cells" is believed to refer to the process globally over a multiple dequeue intervals. In other words, this description of dequeing "cells" merely suggests that multiple cells are dequeued in consecutive processing intervals (or cell-processing periods). As such, the plural expression of the term "cells" in this context should not be interpreted as referring to dequeuing multiple packets. While Applicants readily admit that the language chosen by the drafter in column 7, lines 40-42; and column 9, lines 52-53 is somewhat ambiguous, any ambiguity is negated by the context of the patent specification.

As stated at column 7, lines 46-48 of Petty (in the text explaining the sentence identified by the Examiner), the dequeuing “function dequeues *a cell* from that queue 131, ... , and sends the *dequeued cell*” Both occurrences of the term “cell” are in the singular. Irrespective of the introductory statement, Petty’s detailed description clearly spells out that dequeuing function deals with a single cell at a time.

If it is not empty, the function dequeues *a cell* from that queue 131, at step 604, and sends the dequeued *cell* to physical layer 120, at step 606. Column 7, lines 46-48.
(*emphasis added*)

In addition, the blocks 1004 and 1006 of FIG. 10 of Petty refer to a single cell, namely, “dequeue cell” and “send dequeued cell,” respectively. These portions also evidence that Petty’s system dequeues a single cell at a time.

As mentioned, the process described in the patent is presented in the context of the Generic Cell Rate Algorithm which “transmits one cell from the queue if it meets the GCRA criterion, and the next cell must wait [*if more than one cell is available in the queue of a virtual circuit*] at least until the subsequent processing period to be transmitted.” See column 2, lines 52-58. This ensures a spacing of at least 125 μ s between transmitted ATM cells from any virtual circuit, thereby guaranteeing that the virtual circuit does not exceed its negotiated peak cell rate.

Appellants note that some portions of Petty refer to the “contents” of a queue. However, it is respectfully submitted that nothing in Petty refers to multiple packets as claimed. As appreciated by those skilled in the art, the term “content” is usually used in the plural. However, such a usage does not mean that the contents is actually constituted by multiple packets. Rather, the term “contents” collectively refers to something contained as a whole.

Applicants also note that a prior Action pointed to column 5, line 65 - column 6, line 4 of the Petty patent. The cited portion describes that “one of the queues 131 - the current queue - is emptied by dequeue state machine 141.” However, the term “empty” does not necessarily mean dequeuing all of its contents as claimed. It merely connotes discharging at least some of the contents. In fact, there are many ways to “empty” the current queue including a cell-by-cell basis as suggested by the Petty patent.

One important goal of Petty’s primary goal is to *ensure a minimum interval of 5.785 ms* between successive cells. In other words, the Petty system ensures that a circuit does *not exceed* its peak instantaneous cell rate of *one cell per 5.785 ms*. See,

Petty, column 6, lines 54-58. Such a limitation on an interval between successive cells is necessary in an ATM cell transfer context to avoid congestion.

For at least these reasons, modification of Fan's system/method for dequeuing multiple packets as allegedly taught by Petty is not rendered obvious by a combination of Fan and Petty. Reversal of the § 103(a) rejections is therefore respectfully requested.

Claims 2, 4-8, 10-13, 15, 19-25, 27, 28, and 30

The above arguments for claim 1 apply, *mutatis mutandis*, to claims 2, 4-8, 10, 12, 13, 15, 19-25, 28, and 30, which recite the above-discussed features of independent claim 1.

Claim 3

Claim 3 specifies that "the apparatus is a cable modem termination system." Both of Fan and Petty are related to ATM network systems. Since a cable modem termination system requires a different structural configuration as compared to an ATM network system, mere description regarding ATM network systems does not render use of a cable modem termination system obvious. In this regard, claim 3 provides additional details to the claimed apparatus, making systems of Fan and Petty even less applicable for use in a cable modem termination system context. A separate *prima facie* case of obviousness has not been made. Reversal of the rejection based on a combination of Fan and Petty is respectfully requested.

Claim 9

Claim 9 specifies that "the one or more processors are further configured or designed to transmit, without buffering in a time-based queue, the data or issue grants to transmit data if there is no network traffic shaping delay." Fan is cited as allegedly disclosing this claimed feature. The cited portion, column 7, lines 14-18 of Fan, merely assumes that a stream arrives to an empty queue. However, an empty queue does not mean no network traffic shaping delay. For example, even if the queue is empty, there may be some network traffic delay. Therefore, the above description of Fan does not affect the patentability of claim 9.

Petty has been reviewed and found not to cure the deficiency of Fan. In view of the foregoing, claim 9 provides additional details to the apparatus of claim 1, providing independent grounds for patentability. A separate *prima facie* case of

obviousness has not been made. Reversal of the rejection based on a combination of Fan and Petty is respectfully requested.

Claim 14

Claim 14 specifies that “a policing means for determining whether to buffer the data or grants to transmit data.” Fan is cited as allegedly disclosing this claimed feature. The cited portion, column 17, lines 42-53 of Fan, merely describes updating a timestamp. It does not teach or suggest a policing means for determining whether to buffer the data or grants to transmit data. Therefore, the above description of Fan does not affect the patentability of claim 14.

Petty has been reviewed and found not to cure the deficiency of Fan. In view of the foregoing, claim 14 provides additional details to the apparatus of claim 12, increasing its patentability. A separate *prima facie* case of obviousness has not been made. Reversal of the rejection based on a combination of Fan and Petty is respectfully requested.

Claims 16 and 17

Claim 16 recites “receiving data addressed to the network node prior to determining that transmitting additional data will breach the policy,” and that “the data addressed to the network node is the additional data.” Claim 17 recites “receiving data sent by the network node prior to determining that transmitting additional data will breach the policy,” and that “the data sent by the network node is the additional data.” Fan is cited as allegedly disclosing these claimed features. The cited portion, column 3, lines 35-43 of Fan, merely describes calculation of a timestamp. It does not teach or suggest breaching a policy. Nor does it teach or suggest the claimed timing of receiving data (i.e., prior to determining that transmitting additional data will breach the policy). Therefore, the above description of Fan does not affect the patentability of claim 16.

Petty has been reviewed and found not to cure the deficiency of Fan. In view of the foregoing, claims 16 and 17 recite additional features pertaining to the method of claim 15, thereby providing independent bases for patentability. A separate *prima facie* case of obviousness has not been made. Reversal of the rejection based on a combination of Fan and Petty is respectfully requested.

Claims 18 and 26

Claim 18 recites “calculating a network capacity used by the network node if the additional data was to be transmitted, the calculation being performed prior to determining that transmitting the additional data will breach the policy.” Claim 26 recites limitations corresponding to those of claim 18. Fan is cited as allegedly disclosing this claimed feature. The cited portion, column 16, lines 40-64 of Fan, merely describes calculation of a dynamic rate, and an external excess rate. It does not teach or suggest the claimed timing of calculating a network capacity (i.e., prior to determining that transmitting additional data will breach the policy). Therefore, the above description of Fan does not affect the patentability of claims 18 and 26.

Petty has been reviewed and found not to cure the deficiency of Fan. In view of the foregoing, claims 18 and 26 recite additional features regarding the method of claim 15, thereby providing independent bases for patentability. A separate *prima facie* case of obviousness has not been made. Reversal of the rejection based on a combination of Fan and Petty is respectfully requested.

Conclusion

Appellants have pointed out that the cited references contain insufficient teachings to render the claims *prima facie* obvious. In combination, the references fail to teach or suggest the claimed invention.

In view of the foregoing, it is respectfully submitted that none of the pending claims are rendered unpatentable by the Fan and Petty patents. Accordingly, the pending rejections of all of the claims under 35 U.S.C. § 103 should be reversed.

Respectfully submitted,

BEYER WEAVER LLP

A handwritten signature in black ink, appearing to read "Jeffrey K. Weaver", with a long horizontal flourish extending to the right.

Jeffrey K. Weaver
Reg. No. 31,314

P.O. Box 70250
Oakland, CA 94607-0250
(510) 663-1100

(8) CLAIMS APPENDIX

APPENDIX PENDING CLAIMS

1. (previously presented) An apparatus for controlling data flow through a network, the apparatus comprising:

one or more processors;

memory coupled to at least one of the one or more processors; and

a plurality of time-based queues logically configured on the memory and together defining a period of time with each time-based queue defining a separate increment of time within the period of time, whereby each time-based queue is set to dequeue all of its contents at a separate time, every time that a specified increment of time elapses,

wherein the processor is configured or designed to direct (i) data or (ii) grants to transmit data to particular time-based queues based upon network traffic shaping delays prescribed for the data or grants to transmit the data,

wherein each time-based queue is configured to dequeue more than one packet.

2. (original) The apparatus of claim 1, wherein the apparatus is a router.

3. (original) The apparatus of claim 1, wherein the apparatus is a cable modem termination system.

4. (original) The apparatus of claim 1, wherein the separate increments of time defined by the time-based queues are each of the same length.

5. (original) The apparatus of claim 1, wherein the separate increments of time defined by the time-based queues are configurable.

6. (original) The apparatus of claim 1, wherein the period of time defined by the plurality of time-based queues are configurable.

7. (original) The apparatus of claim 1, wherein the one or more processors are further configured or designed to determine network traffic shaping delay.

8. (original) The apparatus of claim 1, wherein the one or more processors are further configured or designed to discard data or a request to grant transmission of data if a network traffic shaping delay is greater than the period of time defined by the plurality of time-based queues.

9. (original) The apparatus of claim 1, wherein the one or more processors are further configured or designed to transmit, without buffering in a time-based queue, the data or issue grants to transmit data if there is no network traffic shaping delay.

10. (original) The apparatus of claim 1, wherein the one or more processors are further configured or designed to direct network packets of varying sizes to the time-based queues.

11. (original) The apparatus of claim 1, wherein the apparatus is configured or designed to simultaneously buffer, in a single time-based queue, data or grants to transmit data from a plurality of network nodes.

12. (previously presented) An apparatus for controlling data flow through a network, the apparatus comprising:

traffic shaping means for determining how long to buffer data or grants to transmit data; and

buffering means for buffering the data or grants to transmit data in a plurality of time-based queues together defining a period of time, with each time-based queue defining a separate increment of time within the period of time, whereby each time-based queue is set to dequeue all of its contents at a separate time, every time that a specified increment of time elapses,

wherein each time-based queue is configured to dequeue more than one packet.

13. (original) The apparatus of claim 12, wherein the traffic shaping means also directs the data or grant to transmit data to particular time-based queues based upon a determined length of time for buffering.

14. (original) The apparatus of claim 12, further comprising a policing means for determining whether to buffer the data or grants to transmit data.

15. (previously presented) A method of controlling data flow through a network, the method comprising:

determining that transmitting additional data to or from a network node will breach a policy for the network node;

selecting one of a plurality of time-based queues that together define a period of time, with each time-based queue defining a separate increment of time within the time period, whereby each time-based queue is set to dequeue all of its contents at a separate time associated with its increment of time, every time that a specified increment of time elapses; and

buffering the additional data or a grant to transmit the additional data in the selected one of the plurality of time-based queues,

wherein at least some times when de-queuing of all its contents, a time-based queue dequeues more than one packet.

16. (previously presented) The method of claim 15, further comprising receiving data addressed to the network node prior to determining that transmitting additional data will breach the policy, and wherein the data addressed to the network node is the additional data.

17. (previously presented) The method of claim 15, further comprising receiving data sent by the network node prior to determining that transmitting the additional data will breach the policy, and wherein the data sent by the network node is the additional data.

18. (previously presented) The method of claim 15, further comprising calculating a network capacity used by the network node if the additional data was to be transmitted, the calculation being performed prior to determining that transmitting the additional data will breach the policy.

19. (original) The method of claim 15, further comprising determining a delay until the additional data can be transmitted, wherein the determined delay is used to select the time-based queue.

20. (original) The method of claim 19, wherein the time-based queue is selected by matching its time to dequeue with the delay determined for the additional data.

21. (original) The method of claim 15, further comprising:
dequeueing the additional data; and
transmitting the additional data without exceeding the maximum allowed data flow for the network.

22. (previously presented) The method of claim 15, further comprising:
receiving new data that does not form part of the additional data;
determining that transmitting the new data will breach the policy;
determining a delay until the new data can be transmitted without exceeding the maximum allowed data flow for the network node; and
determining that the delay is sufficiently long that the new data is discarded without buffering in the time-based queues.

23. (original) The method of claim 15, wherein the separate increments of time defined by the time-based queues are each of the same size.

24. (original) The method of claim 15, wherein the increments of time defined by the time-based queues are configurable, and wherein the period of time defined by the plurality of time-based queues is configurable.

25. (previously presented) A computer program product comprising a machine-readable medium on which are stored program instructions for controlling data flow through a network, the program instructions comprising:

determining that transmitting additional data to or from a network node will breach a policy for the network node;

selecting one of a plurality of time-based queues that together define a period of time, with each time-based queue defining a separate increment of time within the time period, whereby each time-based queue is set to dequeue all of its contents at a separate time associated with its increment of time, every time that a specified increment of time elapses; and

buffering the additional data or a grant to transmit the additional data in the selected one of the plurality of time-based queues,

wherein at least some times when de-queuing of all its contents, a time-based queue dequeues more than one packet.

26. (previously presented) The computer program product of claim 25, further comprising program instructions for calculating a network capacity used by the network node if the additional data was to be transmitted, the calculation being performed prior to determining that transmitting the additional data will breach the policy.

27. (previously presented) The computer program product of claim 25, further comprising program instructions for:

receiving new data that does not form part of the additional data;

determining that transmitting the new data will breach the policy;

determining a delay until the new data can be transmitted without exceeding the maximum allowed data flow for the network node; and

determining that the delay is sufficiently long that the new data is discarded without buffering in the time-based queues.

28. (previously presented) A computer program product comprising a machine readable medium on which is provided program instructions for controlling data flow through a network, the program instructions comprising:

program code for determining that transmitting additional data to or from a network node will breach a policy for the network node;

program code for selecting one of a plurality of time-based queues that together define a period of time, with each time-based queue defining a separate increment of time within the time period, whereby each time-based queue is set to dequeue all of its contents at a separate time associated with its increment of time, every time that a specified increment of time elapses; and

program code for buffering the additional data or a grant to transmit the additional data in the selected one of the plurality of time-based queues,

wherein at least some times when de-queuing of all its contents, a time-based queue dequeues more than one packet.

29. (canceled)

30. (previously presented) An apparatus for controlling data flow through a network, the apparatus comprising:

means for determining that transmitting additional data to or from a network node will breach a policy for the network node;

means for selecting one of a plurality of time-based queues that together define a period of time, with each time-based queue defining a separate increment of time within the time period, whereby each time-based queue is set to dequeue all of its contents at a separate time associated with its increment of time, every time that a specified increment of time elapses; and

means for buffering the additional data or a grant to transmit the additional data in the selected one of the plurality of time-based queues,

wherein each time-based queue is configured to dequeue more than one packet.

(9) EVIDENCE APPENDIX

None.

(10) RELATED PROCEEDINGS APPENDIX

None.